The USENIX09 class

Stephen Checkoway
`s@cs.ucsd.edu`

Version v1.3, 2011/01/29, Generated July 2, 2016

Abstract

This class is designed to meet the basic requirements for USENIX 2009 paper submission.[1] It should be used in place of the article class.

Contents

1 Introduction

This package is meant to simplify complying with the USENIX 2009 paper submission format by including the appropriate packages and setting document parameters to values appropriate for USENIX 2008 proceedings. USENIX provides an example style file that was designed for USENIX 2005. It contains some of the basic functionality of this class,

---

[1] `http://www.usenix.com/events/usenix09/instrux/details.html`

but much is left up to the user. In addition, the sample file provided does not even make use of all of the functionality provided.

2   Usage

The `usenix09` document class should be used in place of article. It accepts all of the options that article does as well as a number of options to control its behavior. For initial submission, one might use,

> `\documentclass[twocolumn,pagenumbers]{usenix09}`

whereas for camera-ready, the following is more appropriate.

> `\documentclass[twocolumn,footasend]{usenix09}`

In particular, since USENIX requires two columns, the `twocolumn` option should be used. However, since it may be desirable to have a single column, or if two columns are to be achieved some other way, for example using multicol, an option `onecolumn` may be passed. It is an error to not specify either `onecolumn` or `twocolumn`.

Some options like `titlepage` may not work. The `10pt` and `letterpaper` options are passed by default. The various options and their rationales are described next.

2.1   Options

All of the options except for `twocolumn` and `onecolumn` are independent and can be used in any combination.

`onecolumn` Overrides the two column requirement and forces a single column of text. This can be used for initial submission or when the two columns are being set by a package rather than using the `article` document class's option.

`twocolumn` In general, this option should be passed to `usenix09` as USENIX requires two columns. Passing this option will also ensure that the columns are separated by the appropriate amount as specified by USENIX.

`tight` If your document is too long for the page constraints, this option will remove much of the blank space left by `\maketitle` as well as supplying the `compact` option to titlesec. The macros in titlesec can be manipulated to change the spacing whether or not the `tight` option is passed.

`titlecompat` This option emulates the spacing of the usenix package. It overrides `tight`, but only for `\maketitle`.

`footasend` USENIX requires that endnotes be used in place of footnotes. This option simplifies the conversion process by changing `\footnote` to `\endnote` and using `\theendnotes` at the very end of the document. Without the option, `\endnote` should be used and `\theendnotes` should be placed at the end manually.

`pagenumbers` USENIX requires that initial submissions contain page numbers but that the camera-ready submission contain no page numbers.

`nofonts` By default, usenix09 loads the mathptmx, helvet, and courier packages to set the roman and math fonts to times, the sans serif font to a scaled down Helvetica, and the typewriter font to Courier. Pass the `nofonts` option if you wish to set the fonts yourself. Passing this option also sets `nofontenc`.

`nofontenc` Pass this option if you do not want T1 fonts used. I can't think of a reason to use this, but since this isn't required by USENIX, I thought I'd give an option for it.

`noflushend` The USENIX specifications do not say if the columns on the final page should be the same length or not. I think it looks better when they are the same length and thus the flushend package is used by default. If you prefer to handle this yourself, pass this option.

2.2 Authors

USENIX is particular about the format of the authors. That is, each author's name should be set in 12 pt roman font while the institution should be set in 12 pt italics font. The standard `\author` macro does not accommodate this without resorting to various hacks. For example, if you want multiple rows of authors, then rather than separate each author by `\and`, you have to use the fact that `\author` is typeset in a `tabular` and use two authors in a column separated by `\and` followed by another two authors, and so forth. This doesn't scale particularly well and it requires a lot of manual futzing.

`\addauthor`   To avoid the problems with `\author`, a new command

$$\verb|\addauthor[|\langle\textit{footnote}\rangle\verb|]{|\langle\textit{name}\rangle\verb|}{|\langle\textit{institution}\rangle\verb|}|$$

is provided. It should be used once for each author. The ⟨*footnote*⟩ optional argument can be used to specify a footnote for the author's name—as specified by ⟨*name*⟩. The ⟨*institution*⟩ argument will be below the author and in italics.

Using `\footnote` or `\thanks` in the ⟨*name*⟩ or ⟨*institution*⟩ arguments directly is supported, but the order of the footnotes will not be what you might expect. Namely, they will appear after footnotes in the title as well as after the ⟨*footnote*⟩ footnotes. Worse, the order will not be left-to-right, top-to-bottom (unless there is only a single row of authors). Basically, just don't do that. Use ⟨*footnote*⟩ instead.

`\author`   In addition to `\addauthor`, `\author` is provided as an alias for `\addauthor`.
`\and`   The `\and` macro is no longer relevant and its use is an error.
`\authorsep`   Two parameters control the horizontal and vertical spacing of the authors. The hor-
`\authorskip`   izontal spacing is controlled by `\authorsep` and the default value is that used by the article class's `\and` macro. The vertical spacing is controlled by `\authorskip`. The default value is probably a reasonable one. Neither of these need to be changed, but both can be.
`\authorcolumns`   The number of columns of authors (and hence the number of rows) is controled by the `\authorcolumns` parameter. If it is positive, then the author list will be typeset

using that many columns. If the number of columns specified cannot fit on the page, then the columns will *not* wrap. Instead, the authors will extend into the margins.

If \authorcolumns is nonpostive, then the default number of columns will be chosen. If there are one to four authors (inclusive), then they will be in a single row. Five and six authors use two rows of at most three columns. Seven and eight authors use two rows of at most four columns. Nine authors use three rows of three columns. Ten or more authors are typeset in rows of at most four columns. In all cases, only the last row may have fewer than the maximum number of columns.

If the authors names and institutions are very long, they may not fit into the number of columns chosen by default. In this case, \authorcolumns should be manually set or \authorsep can be decreased.

\authornewline \institutionnewline   Additional lines can be added under the author or institution by using \authornewline or \institutionnewline, but since the author list is typeset in columns rather than rows, each author in a row should have the same number of lines, even if they are blank.

In some cases with many authors and few institutions, it takes up less space to list all authors from the first institution together with the institution and then below it all authors from the second institution and so on. In this case, \authorcolumns should be set to 1 and all authors for a single institution should be given in the {⟨*name*⟩} parameter to \addauthor. Extra lines of authors for a given institution can be given with \authornewline.

3 Implementation

3.1 Initial code

First we add the package identification and then we declare some conditionals which we will set with options.

```
1 \NeedsTeXFormat{LaTeX2e}[1999/12/01]
2 \ProvidesClass{usenix09}
3     [2011/01/29 v1.3 USENIX09 formatting]
```

`\ifusenix@onecolumn`   One column flag.

```
4 \newif\ifusenix@onecolumn \usenix@onecolumnfalse
```

`\ifusenix@nofonts`   Don't load any fonts.

```
5 \newif\ifusenix@nofonts \usenix@nofontsfalse
```

`\ifusenix@nofontenc`   No fontenc flag.

```
6 \newif\ifusenix@nofontenc \usenix@nofontencfalse
```

`\ifusenix@tight`   Tight flag.

```
7 \newif\ifusenix@tight \usenix@tightfalse
```

`\ifusenix@titlecompat`   `\maketitle` compatibility with usenix package.

```
8 \newif\ifusenix@titlecompat \usenix@titlecompatfalse
```

`\ifusenix@pagenumbers`   Page numbers flag.

```
9 \newif\ifusenix@pagenumbers \usenix@pagenumbersfalse
```

`\ifusenix@noflushend`   No flushend flag.

```
10 \newif\ifusenix@noflushend \usenix@noflushendfalse
```

3.2 Options

We want to support several options which change the way the document is laid out.

The twocolumn option is a global option. USENIX 2009 requires documents to be typeset in two columns. Pass onecolumn to allow single column text.

```
11 \DeclareOption{onecolumn}{\usenix@onecolumntrue}
```

Set the section spacing to be very tight for long papers.

```
12 \DeclareOption{tight}{\usenix@tighttrue\PassOptionsToPackage{compact}{titlesec}}
```

Turn on `\maketitle` compatibility with usenix package.

```
13 \DeclareOption{titlecompat}{\usenix@titlecompattrue}
```

Make `\footnote` synonymous with `\endnote`.

```
14 \DeclareOption{footasend}{
15     \AtBeginDocument{\let\footnote\endnote}
16     \AtEndDocument{\theendnotes}}
```

Do not set the fonts at all.

```
17 \DeclareOption{nofonts}{\usenix@nofontstrue\ExecuteOptions{nofontenc}}
```

Do not load the fontenc package.

```
18 \DeclareOption{nofontenc}{\usenix@nofontenctrue}
```

Add page numbers.

```
19 \DeclareOption{pagenumbers}{\usenix@pagenumberstrue}
```

Do not flush end.

```
20 \DeclareOption{noflushend}{\usenix@noflushendtrue}
```

Pass all unknown options to the article class.

```
21 \DeclareOption*{\PassOptionsToClass{\CurrentOption}{article}}
```

Process those options! (Use `\relax` to prevent the search for an asterisk.) Load the `article` class.

```
22 \ProcessOptions\relax
23 \LoadClass[10pt,letterpaper]{article}
```

3.3 Loading the packages

We need to load a number of packages to ensure that we conform. But before we do that, load the `fixltx2e` package to fix the float placement, among other things.

```
24 \RequirePackage{fixltx2e}
```

Flush end?

```
25 \ifusenix@noflushend\else
26     \RequirePackage{flushend}
27 \fi
```

We need to set the roman and math fonts to Times, the sans serif font to Helvetica, and the typewriter font to Courier.

```
28 \ifusenix@nofonts\else
29     \RequirePackage{mathptmx}
30     \RequirePackage[scaled=.92]{helvet}
31     \RequirePackage{courier}
32 \fi
```

If nofontenc was given, do not load the package. Otherwise, use T1 fonts.

```
33 \ifusenix@nofontenc\else
34     \RequirePackage[T1]{fontenc}
35 \fi
```

We also want the geometry, titlesec, caption, and endnotes packages to set the page layout, section heading, captions, and endnotes appropriately.

```
36 \RequirePackage[letterpaper,margin=1in,nohead,columnsep=.25in]{geometry}
37 \RequirePackage{titlesec}
38 \RequirePackage[margin=10pt,font=small,labelfont=bf]{caption}
39 \RequirePackage{endnotes}
40 \let\enotesize\normalsize
41 \def\enoteformat{%
```

```
42      \rightskip\z@
43      \leftskip\z@
44      \leavevmode\llap{\makeenmark}%
45 }
```

3.4   Satisfying the format

If we don't have page numbers, then we don't need space for them.

```
46 \ifusenix@pagenumbers\else
47 \geometry{nofoot}
48 \fi
```

\usenix@twocolumn   Make sure we're either in `twocolumn` or `onecolumn`. Camera ready requires a column separation of 0.25 in.

```
49 \begingroup
50 \def\usenix@twocolumn{
51     \ifusenix@onecolumn
52         \ClassError{usenix09}{Cannot specify both twocolumn
53         and onecolumn.}{At most one of twocolumn and onecolumn
54         may be specified.}
55     \fi}
```

\usenix@onecolumn

```
56 \def\usenix@onecolumn{
57     \ifusenix@onecolumn\else
58         \ClassError{usenix09}{USENIX 2009 requires 2
59         columns.\MessageBreak
60         Pass option onecolumn to override.}{Single column
61         requires the use of option onecolumn.}
62     \fi}

63 \if@twocolumn
64     \usenix@twocolumn
65     \setlength\columnsep{0.25in}
66 \else
67     \usenix@onecolumn
68 \fi
69 \endgroup
```

Do we want page numbers?

```
70 \ifusenix@pagenumbers
71     \pagestyle{plain}
72 \else
73     \pagestyle{empty}
74 \fi
```

\maketitle   We need to change the title layout. This is mostly copied from `classes.dtx`.

```
75 \renewcommand\maketitle{\par
76     \begingroup
```

```
77    \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
78    \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}}%
79    \long\def\@makefntext##1{\parindent1em\noindent
80        \hb@xt@1.8em{%
81            \hss\@textsuperscript{\normalfont\@thefnmark}}##1}%
82    \if@twocolumn
83        \twocolumn[\@maketitle]%
84    \else
85        \newpage
86        \global\@topnum\z@
87        \@maketitle
88    \fi
```

Make sure we don't use page numbers unless they were requested.

```
89    \ifusenix@pagenumbers
90        \thispagestyle{plain}%
91    \else
92        \thispagestyle{empty}%
93    \fi
94    \@thanks
95    \endgroup
96    \setcounter{footnote}{0}%
```

Now clear out a number of macros related to the title generation.

```
97    \begingroup
98    \def\r##1{\global\let##1\@undefined}%
99    \global\let\@thanks\@empty
100   \global\let\@author\@empty
101   \global\let\@date\@empty
102   \global\let\@title\@empty
103   \r\thanks
104   \r\maketitle
105   \r\@maketitle
106   \r\title
107   \r\author
108   \r\date
109   \r\and
110   \r\addauthor
111   \r\usenix@authorname
112   \r\usenix@institutionname
113   \r\usenix@extract
114   \r\usenix@temp
115   \r\usenix@layoutauthorcolumn
116   \r\usenix@l@youtauthorcolumn
117   \r\usenix@layoutauthors
118   \r\authornewline
119   \r\institutionnewline
120   \global\usenix@currauthor\z@
121   \loop\ifnum\count@<\usenix@numauthors
122       \expandafter
123       \r\csname usenix@author\the\usenix@currauthor\endcsname
```

8

```
124              \global\advance\usenix@currauthor\@ne
125      \repeat
126      \endgroup
127 }
```

\@maketitle   No date is displayed, the title is in 14 pt **bold**. The author names and affiliations are
handled by \usenix@layoutauthors, below.

```
128 \renewcommand\@maketitle{%
129     \newpage
130     \ifusenix@titlecompat
131         \vbox to2.5in{
132             \vskip2em plus1fill
133             \begin{center}%
134                 \let\footnote\thanks
135                 {\Large\bfseries\@title\par}%
136                 \vskip.375in minus.3in
137                 {\large\lineskip.5em \usenix@layoutauthors}%
138             \end{center}%
139             \par
140             \vfill
141         }%
142     \else % usenix@titlecompat
143         \null
144         \ifusenix@tight\else
145             \vskip2em
146         \fi
147         \begin{center}%
148             \let\footnote\thanks
149             {\Large\bfseries\@title\par}%
150             \ifusenix@tight
151                 \vskip.75em
152             \else
153                 \vskip1.5em
154             \fi
155             {\large\lineskip.5em \usenix@layoutauthors}%
156             \ifusenix@tight
157                 \vskip.5em
158             \else
159                 \vskip1em
160             \fi
161         \end{center}%
162         \par
163         \ifusenix@tight
164             \vskip.75em
165         \else
166             \vskip1.5em
167         \fi
168     \fi % usenix@titlecompat
169 }
```

9

3.5 Handling multiple authors

The author layout is is a bit tricky. We want to emulate the behavior of `\and` between authors, but we also need to fulfill the requirements of USENIX, namely the author name needs to be in roman font and the institution must be in italics.

`\and` To handle an arbitrary number of authors, we remove the `\and` macro and replace it with `\addauthor` which is to be called multiple times, one for each author.

```
170 \renewcommand\and{\ClassError{usenix09}{\string\and\ is not supported.
171 Use \string\addauthor.}{\string\and\ does not work with usenix09.}}
```

`\addauthor` Save the author for later.

```
172 \newcommand\addauthor[3][]{%
173     \expandafter\def\csname usenix@author\the\usenix@numauthors\endcsname{%
174         {#1}{#2}{#3}}
175     \global\advance\usenix@numauthors\@ne
176 }
```

`\author` Make `\author` an alias for `\addauthor`.

```
177 \global\let\author\addauthor
```

Set the three user-accessible spacing registers to reasonable values.

`\authorsep` This controls horizontal separation.

```
178 \newskip\authorsep \authorsep1em\@plus.17fil
```

`\authorskip` This controls vertical skip.

```
179 \newskip\authorskip \authorskip.5em
```

`\authorcolumns` Let the user specify the number of columns desired.

```
180 \newcount\authorcolumns \authorcolumns\z@
```

Now set up some internally needed counters and macros.

```
181 \newcount\usenix@numauthors
182 \newcount\usenix@currauthor
183 \global\usenix@numauthors\z@
184 \providecommand\@firstofthree[3]{#1}
185 \providecommand\@secondofthree[3]{#2}
186 \providecommand\@thirdofthree[3]{#3}
```

We need to be able to extract the three components of the current author. The argument should be one of `\@firstofthree`, `\@secondofthree`, or `\@thirdofthree`. The three `\expandafter`s cause the constructed control sequence to be expanded before the argument extracts the individual component. Note that there is zero error checking here.

```
187 \newcommand\usenix@extract[1]{\expandafter\expandafter\expandafter
188     #1\csname usenix@author\the\usenix@currauthor\endcsname}
```

To simplify matters, provide a single macro that expands to the author's name with a footnote if desired in roman as well as one for the institution.

```
189 \newcommand\usenix@authorname{%
190     \protected@edef\usenix@temp{\usenix@extract\@firstofthree}%
191     \begingroup
192     \upshape
193     \usenix@extract\@secondofthree
194     \endgroup
195     \ifx\usenix@temp\@empty\else
196         \footnotemark[\usenix@temp]%
197     \fi
198 }
199 \newcommand\usenix@institutionname{\usenix@extract\@thirdofthree}
```

`\usenix@layoutauthors` This is the hard part. We want a somewhat unified style for the authors. The defaults should be sensible, but can be overwritten by changing `\authorcolumns`.

```
200 \newcommand\usenix@layoutauthors{%
```

We need to process the footnotes at this point to ensure that they follow any in the title.

```
201     \global\usenix@currauthor\z@
202     \loop\ifnum\usenix@currauthor<\usenix@numauthors
203         \protected@edef\usenix@temp{\usenix@extract\@firstofthree}%
204         \ifx\usenix@temp\@empty\else
205             \stepcounter{footnote}%
206             \protected@xdef\@thefnmark{\thefootnote}%
207             \protected@xdef\@thanks{\@thanks
208                 \protect\footnotetext[\the\c@footnote]{\usenix@temp}}%
209             \def\usenix@temp{\the\c@footnote}%
210         \fi
```

We want to replace the first component of the author with the footnote number. We do this by expanding the `\csname` and then the resultant control sequence, and then the `\@gobble` to remove the first component and stuff the remainder into the token register. Then the author is redefined to be the new triple.

```
211         \toks@\expandafter\expandafter\expandafter\expandafter\expandafter
212             \expandafter\expandafter{\expandafter\expandafter\expandafter
213             \@gobble\csname usenix@author\the\usenix@currauthor\endcsname}
214         \expandafter\edef
215             \csname usenix@author\the\usenix@currauthor\endcsname{%
216                 {\usenix@temp}\the\toks@}
217         \global\advance\usenix@currauthor\@ne
218     \repeat
```

Get on with laying out the authors in columns.

```
219     \ifnum\usenix@numauthors=\z@
220         \ClassError{usenix09}{At least one author is required.}{Use
221         \string\addauthor\ to add an author.}
222     \fi
223     \ifnum\authorcolumns>\z@
224         \ifnum\authorcolumns>\usenix@numauthors
```

```
225          \authorcolumns\usenix@numauthors
226       \fi
227    \else
228       \authorcolumns
229       \ifcase\usenix@numauthors
230       \or \@ne %1
231       \or \tw@ %2
232       \or \thr@@ %3
233       \or 4 %4
234       \or \thr@@ %5
235       \or \thr@@ %6
236       \or 4 %7
237       \or 4 %8
238       \or \thr@@ %9
239       \else 4 % How many authors do you have??
240       \fi
241    \fi
```

Now that `\authorcolumns` has been set to a reasonable value, we can perform the actual layout. This works by computing $\lfloor numauthors/authorcolumns \rfloor \cdot authorcolumns$ and laying out that many authors in the requested number of columns. If *authorcolumns* minus the remaining number of authors is even, then layout those authors in the columns that are in the middle. Otherwise, just lay out the authors.

```
242    \begingroup
243    \itshape
244    \count@\usenix@numauthors
245    \divide\count@\authorcolumns
246    \multiply\count@\authorcolumns
247    \count\z@\authorcolumns
248    \advance\count\z@-\usenix@numauthors
249    \advance\count\z@\count@
250    \ifodd\count\z@
251        \count\z@\authorcolumns
252    \else
253        \divide\count\z@\tw@
254    \fi
255    \count\tw@\count@
256    \count4\z@
257    \hb@xt@\textwidth{\hfil
258    \loop\ifnum\count4<\authorcolumns
259        \ifnum\count4>\z@
260            \hskip\authorsep\relax
261            \fi
262        \ifnum\count\z@>\z@
263            \usenix@layoutauthorcolumn{\count4}\usenix@numauthors
264            \advance\count\z@\m@ne
265        \else
266            \usenix@layoutauthorcolumn{\count4}{\count\tw@}%
267            \advance\count\tw@\@ne
268        \fi
```

12

```
269          \advance\count4\@ne
270      \repeat
271      \global\usenix@currauthor\count\tw@
272      \hfil}%
273      \ifnum\usenix@currauthor<\usenix@numauthors
274          \par\vskip\authorskip\relax
275          \hb@xt@\textwidth{\hfil
276          \loop\ifnum\usenix@currauthor<\usenix@numauthors
277              \begin{tabular}[t]{c}%
278              \usenix@authorname\\
279              \usenix@institutionname
280              \end{tabular}%
281              \global\advance\usenix@currauthor\@ne
282              \ifnum\usenix@currauthor<\usenix@numauthors
283                  \hskip\authorsep\relax
284              \fi
285          \repeat
286          \hfil}
287      \fi
288      \endgroup
289 }
```

\usenix@layoutauthorcolumn At last, the code for handling a single column. The first argument is the starting point, the second is an additional author to place in the column for the final row. The column is layed out by starting at the first argument and then striding through the authors by \authorcolumns until \count@ is reached. Then the second author is layed out, assuming it is in range.

```
290 \newcommand\usenix@layoutauthorcolumn[2]{%
291      \global\usenix@currauthor#1
292      \begin{tabular}[t]{c}%
293      \usenix@l@youtauthorcolumn
294      \global\usenix@currauthor#2
295      \ifnum\usenix@currauthor<\usenix@numauthors
296          \\[\authorskip]%
297          \usenix@authorname\\
298          \usenix@institutionname
299      \fi
300      \end{tabular}%
301 }
302 \newcommand\usenix@l@youtauthorcolumn{%
303      \usenix@authorname\\
304      \usenix@institutionname
305      \global\advance\usenix@currauthor\authorcolumns
306      \ifnum\usenix@currauthor<\count@
307          \\[\authorskip]%
308          \expandafter\usenix@l@youtauthorcolumn
309      \fi
310 }
```

Add the newline helpers.

`\authornewline`
`\institutionnewline`

```
311 \newcommand*\authornewline{%
312     \unskip
313     \endgroup
314     \tabularnewline
315     \begingroup
316     \upshape
317     \ignorespaces
318 }
319 \newcommand*\institutionnewline{%
320     \unskip
321     \tabularnewline
322     \ignorespaces
323 }
```

3.6 Setting up the section headers

USENIX requires that the section heading be in 12 pt **bold**. There is no requirement on the others, so just make them smaller.

```
324 \titleformat\section{\bfseries\large}{\thetitle}{1em}{}{}
325 \titleformat\subsection{\bfseries\fontsize\@xipt{13}\selectfont}{\thetitle}{1em}{}{
326 \titleformat\subsubsection{\bfseries}{\thetitle}{1em}{}{}
327 \endinput
```

Change History

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.